PHILOSOPHICAL
TRANSACTIONS A

rsta.royalsocietypublishing.org

Research

Article submitted to journal

Subject Areas:
Computer modelling and simulation; artificial intelligence; climatology.

Keywords:
Numerical modelling; data assimilation; machine learning.

Author for correspondence:
Julien Brajard
e-mail: julien.brajard@nersc.no

Combining data assimilation and machine learning to infer unresolved scale parametrisation.

Julien Brajard[1,2], Alberto Carrassi[3,4], Marc Bocquet[5] and Laurent Bertino[1]

[1]Nansen Center (NERSC), 5006, Bergen, Norway
[2]Sorbonne University, Paris, France
[3]Department of Meteorology, University of Reading and NCEO, United-Kingdom
[4]Mathematical Institute, University of Utrecht, The Netherlands
[5]CEREA, joint laboratory École des Ponts ParisTech and EDF R&D, Université Paris-Est, France

In recent years, machine learning (ML) has been proposed to devise data-driven parametrisations of unresolved processes in dynamical numerical models. In most cases, the ML training leverages high-resolution simulations to provide a dense, noiseless target state. Our goal is to go beyond the use of high-resolution simulations and train ML-based parametrisation using direct data, in the realistic scenario of noisy and sparse observations.

The algorithm proposed in this work is a two-step process. First, data assimilation (DA) techniques are applied to estimate the full state of the system from a truncated model. The unresolved part of the truncated model is viewed as a model error in the DA system. In a second step, ML is used to emulate the unresolved part, a predictor of model error given the state of the system. Finally, the ML-based parametrisation model is added to the physical core truncated model to produce a hybrid model.

The DA component of the proposed method relies on an ensemble Kalman filter while the ML parametrisation is represented by a neural network. The approach is applied to the two-scale Lorenz model and to MAOOAM, a reduced-order coupled ocean-atmosphere model. We show that in both cases the hybrid model yields forecasts with better skill than the truncated model. Moreover, the attractor of the system is significantly better represented by the hybrid model than by the truncated model.

THE ROYAL SOCIETY
PUBLISHING

1. Introduction

The Earth climate system is one example of a natural system that is reasonably well represented through known physical laws and that has been intensively observed for decades (see, *e.g.*, [1]). Physical laws, in the form of ordinary (ODEs) or partial differential equations (PDEs), are implemented through numerical models providing the time evolution of the system's state. Although weather and climate predictions have constantly improved, and will likely continue to do so, uncertainties will ineluctably remain. Those usually fall into two major classes: (i) the internal variability driven by the sensitivity to the initial conditions, and (ii) the model errors. The former has to do with the amplification of the initial condition error and arises even in perfect models - it is mitigated by using data assimilation (DA) [2]. The latter is present even if one would perfectly know the initial conditions and has to do with the incorrect and/or incomplete representation of the laws governing the system. The two sources of errors are inevitably entangled and it is difficult to separate them in practice.

Machine learning (ML) was recently shown to be effective in reducing model error, in particular that originating from unresolved scales. This has been achieved using two approaches. The first consists in learning a subgrid parameterisation of a model from existing physics-based expensive parametrisation schemes [3,4], or from the differences between high- and low-resolution simulations [5–8]. In those approaches, the unresolved part of the model is represented by a ML process while the core of the model is derived from ODEs. The second approach is to emulate the entire model using observations. With spatially dense and noise-free data, this approach has been based on sparse regression [9], echo state networks [10,11], recurrent neural networks [12], residual neural network [13] or convolutional neural networks [14,15]. The challenging problem of partial and/or noisy observations has been addressed using dedicated NN architecture [16] or in combination with data assimilation methods [17–21].

This work presents a new method to obtain a data-driven parameterisation of a model's unresolved scale. In particular, we aim at producing a hybrid model combining the physics-based core (encoding the best of our knowledge of the resolved scales physics) with the data-driven parameterisation. By leveraging the use of DA, our method efficiently handles noisy and sparse observations.

2. Objectives and definitions

(a) Statement of the problem

We consider an autonomous chaotic dynamical system, seen as our reference "truth", represented by the ODE

$$\frac{\mathrm{d}\mathbf{z}(t)}{\mathrm{d}t} = f(\mathbf{z}(t)), \tag{2.1}$$

with $\mathbf{z}(t) \in \mathbb{R}^{N_z}$ being the system's state at time t. From an arbitrary state $\mathbf{z}(t)$ on the system's attractor the model can be integrated forward for one time step δt, to get:

$$\mathbf{z}_{\delta t} = \mathcal{M}(\mathbf{z}, \delta t), \tag{2.2}$$

where $\mathbf{z}_{\delta t} = \mathbf{z}(t + \delta t)$.

Let us formally define a projection operator $\mathit{\Pi} : \mathbb{R}^{N_z} \mapsto \mathbb{R}^{N_x}$ such as $\mathbf{x} = \mathit{\Pi}(\mathbf{z})$, with $\mathbf{x}$ being the projection of the full state into a reduced dimension state: $N_x < N_z$. For example, $\mathit{\Pi}$ can be a subsampling operator retaining only a subset of $\mathbf{z}$ or a downscaling operator from a high-resolution state to a lower resolution. From Eq. 2.2, we also define $\mathbf{x}_{\delta t} = \mathit{\Pi}(\mathbf{z}_{\delta t})$.

Consider now a scale-truncated model, $\mathcal{M}^{\mathrm{r}}$, that provides an imperfect description of Eq. (2.2) in the reduced space $\mathbb{R}^{N_x}$,

$$\mathbf{x}^{\mathrm{r}}_{\delta t} = \mathcal{M}^{\mathrm{r}}(\mathbf{x}^{\mathrm{r}}, \delta t), \tag{2.3}$$

where the superscript r stands for "resolved". When initialised from the projection of the truth into the reduced space (*i.e.* no initial condition error: $\mathbf{x}^{\mathrm{r}} = \mathbf{x}$), the difference between the 1-time step predictions from Eqs. (2.2) and (2.3) defines the model error due to the neglected scales in the resolved model

$$\boldsymbol{\epsilon}^{\mathrm{m}}(\mathbf{z}, \delta t) = \mathbf{x}_{\delta t} - \mathbf{x}^{\mathrm{r}}_{\delta t} = \Pi \circ \mathcal{M}(\mathbf{z}, \delta t) - \mathcal{M}^{\mathrm{r}}(\Pi(\mathbf{z}), \delta t). \tag{2.4}$$

The objective of this work is to complement the resolved model (2.3) by an empirical representation of the unresolved scales based on a neural network (NN) trained on incomplete and noisy data. We will hereafter denote the NN-based unresolved scale representation by $g(\mathbf{x}, \boldsymbol{\theta})$. The vector $\boldsymbol{\theta}$ is the set of trainable parameters of the NN and is determined by minimising the loss function

$$L(\boldsymbol{\theta}) = \mathbb{E}_{\mathbf{z}\sim p_Z}\left[g(\Pi(\mathbf{z}), \boldsymbol{\theta}) - \boldsymbol{\epsilon}^{\mathrm{m}}(\mathbf{z}, \delta t)\right]^2, \tag{2.5}$$

where p_Z is the invariant probability density function (assumed to exist) on the attractor.

We construct the hybrid model $\mathcal{M}^{\mathrm{h}}$, parametrised by $\boldsymbol{\theta}$, such that

$$\mathbf{x}^{\mathrm{h}}_{\delta t} = \mathcal{M}^{\mathrm{r}}(\mathbf{x}, \delta t) + g(\mathbf{x}, \boldsymbol{\theta}) = \mathcal{M}^{\mathrm{h}}(\mathbf{x}, \delta t). \tag{2.6}$$

Our objective is to optimally estimate the parameters $\boldsymbol{\theta}$ so that the hybrid model is the most accurate representation of the true underlying dynamics.

Apart from trivial cases, the loss function in Eq. (2.5) cannot be computed: $\mathbf{x}_{\delta t}$ is unknown and p_Z is generally intractable. Assuming ergodicity of the true dynamics, we can however estimate it by a Monte-Carlo approach such that:

$$L(\boldsymbol{\theta}) \approx \frac{1}{K}\sum_{k=0}^{K-1}\left[g(\mathbf{x}_k, \boldsymbol{\theta}) - \boldsymbol{\epsilon}^{\mathrm{m}}(\mathbf{z}_k, \delta t)\right]^2, \tag{2.7}$$

where $\mathbf{z}_{0:K-1} = \{\mathbf{z}_0, \mathbf{z}_1, \cdots, \mathbf{z}_{K-1}\}$, $\mathbf{z}_k = \mathbf{z}(t_k)$, is a set of samples of p_Z, typically extracted from a time series of modelled state variables and $\mathbf{x}_k = \Pi(\mathbf{z}_k)$. Such samples are usually not independent (due to the underlying dynamics being deterministic), and only provide a biased approximation of p_Z. Furthermore, the need to sample the whole attractor implies treating time series significantly longer than the decorrelation time (*i.e.* K very large in general) .

(b) Framework of the study

The loss function, Eq. (2.7), cannot be minimised directly because some of its key entries are unavailable, in particular, obviously, the true process Eq. (2.2) and the time series $\mathbf{z}_{0:K-1}$. The available terms in the loss function are:

The truncated model. The truncated model, our best available knowledge about the true process, is usually very complex (high dimensional, nonlinear and with diagnostic variables). Hence, we shall assume that its gradient, $\nabla_{\mathbf{x}}\mathcal{M}^{\mathrm{r}}$, cannot be computed analytically. We will thus focus on developing a (model) adjoint-free approach that is more flexible and suitable to high dimensional nonlinear scenarios where deriving and maintaining an adjoint model is a difficult and costly task [22]. When the gradient can be computed, other efficient methods exist [21].

Observations. Observations are incomplete and noisy and are obtained through:

$$\mathbf{y}_k = \mathbf{H}_k\mathbf{x}_k + \boldsymbol{\epsilon}^{\mathrm{o}}_k, \tag{2.8}$$

where $\mathbf{x}_k$ is the true state in the reduced space, $\mathbf{y}_k \in \mathbb{R}^{N_y}$ and $\mathbf{H}_k \in \mathbb{R}^{N_y \times N_x}$ are the observation vector and operator respectively at t_k, while $\boldsymbol{\epsilon}^{\mathrm{o}}_k$ is the observation error, assumed to be uncorrelated in time and normally distributed with mean 0 and a variance-covariance matrix $(\sigma^{\mathrm{o}})^2\mathcal{I}_{N_y}$, where $\mathcal{I}_{N_y}$ is the identity matrix of size $N_y \times N_y$ and σ^{o} is the standard deviation. For simplicity, the observation error standard deviation is taken constant and the observation operator linear. Both assumptions can be relaxed without major drawbacks even if it can induce practical difficulties. The ideal, most favourable, situation in which the full system's state is observed in the reduced space with no error is referred as the "perfect observation" case: $\mathbf{y}_k = \mathbf{x}_k$. For convenience, we further assume that observations are available regularly at multiples of the model time step such that $\Delta t = t_{k+1} - t_k = N_c \delta t$, $N_c \in \mathbb{N}^*$. This also accounts for the fact that, in general, the observation sampling period is longer than the integration time step of the numerical model.

3. Method

(a) Loss function approximation

Let us assume that an estimation of $\mathbf{x}_k$ ($k \in \{0, \cdots, K\}$) is available at observation times, every $\Delta t = N_c \delta t$, so that

$$\mathbf{x}_{k+1} = \Pi \circ \mathcal{M}^{(N_c)}(\mathbf{z}_k, \delta t), \tag{3.1}$$

where $\mathcal{M}^{(N_c)}$ is the composition of the model N_c times, and similarly for the truncated model, $\mathbf{x}^{\mathrm{r}}_{k+1} = \mathcal{M}^{\mathrm{r}(\mathrm{N_c})}(\mathbf{x}_k, \delta t)$. Since observations are not available at each time step ($N_c > 1$), the model error $\boldsymbol{\epsilon}^{\mathrm{m}}$ is not known at each time step neither, and the loss function Eq. (2.7) cannot be exactly computed. In the following, we will present two key simplifying assumptions that will lead to a tractable approximation of the loss function.

The first consists in assuming Δt to be short enough so that the state evolved by the truncated model is independent of the model error due to the unresolved scale so that the model error is an additive term to the truncated model forecast after a Δt time integration. The second in that the variability of $\boldsymbol{\epsilon}^{\mathrm{m}}$ is small within Δt. The combination of these two hypotheses is known as the *linear superposition assumption*, and can be formalised as:

$$\mathbf{x}_{k+1} \approx \mathbf{x}^{\mathrm{r}}_{k+1} + N_c \times \boldsymbol{\epsilon}^{\mathrm{m}}(\mathbf{z}_k, \delta t). \tag{3.2}$$

Note that the same approximation was made in a similar setting by [8].

The optimisation can now be performed using the approximate loss function:

$$L(\boldsymbol{\theta}) \approx L^{\mathrm{a}}(\boldsymbol{\theta}) = \frac{1}{K N_c^2} \sum_{k=0}^{K-1} \left[N_c g(\mathbf{x}_k, \boldsymbol{\theta}) - (\mathbf{x}_{k+1} - \mathbf{x}^{\mathrm{r}}_{k+1}) \right]^2. \tag{3.3}$$

The modified loss function, Eq. (3.3) can be computed without knowing the full state $\mathbf{z}_k$ but only its projection $\mathbf{x}_k$. L^{a} can be minimised using a gradient descent algorithm as long as the gradient of $g(\mathbf{x}_k, \boldsymbol{\theta})$ can be computed, which is standard for any neural network library.

(b) Description of the algorithm

In order to minimise the loss function defined in Eq. (3.3), a sequence of $\mathbf{x}_{0:K}$ has to be available. Two cases are considered: the first is the aforementioned "perfect observations" case, in which we have a complete and noise-free sequence of a state variable $\mathbf{x}_k$ in the reduced space. This ideal situation will set the upper-bound performance in the algorithm evaluation that follows. The second case is the more realistic case of noisy (yet unbiased) and possibly incomplete observations. Here, a complete sequence $\mathbf{x}_{0:K}$ is obtained by processing the incomplete and noisy observations using DA [2]: the observations $\mathbf{y}_k$ are combined with the forecast from the truncated model $\mathbf{x}^{\mathrm{r}}_k$ in order to provide the analysed state vector $\mathbf{x}^{\mathrm{a}}_k$. The DA method used in this work is the Finite-Size Ensemble Kalman Filter (EnKF-N) [23] implemented in the DAPPER framework [24]. Even if the proposed algorithm is general and suitable for any DA algorithm, the EnKF-N has been chosen because of its efficiency. In particular, the inflation factor, a needed fix to mitigate the impact of sampling errors in the ensemble-based DA methods, is automatically estimated (thus avoiding long tuning). This inflation factor accounts also implicitly and partially for the effect of the model error.

The correction $\mathbf{x}_k^{\mathrm{a}} - \mathbf{x}_k^{\mathrm{r}}$ made by DA is called analysis increment and was used to estimate model error due to unresolved scales in sequential DA in [25,26]. An analysis increment is composed of a correction both for the model error (which is what the hybrid model aims at estimating) and for the initial conditions error (which cannot be represented by the hybrid model). There is also an additional uncertainty due to the observation errors. The initial conditions and the observations errors are two sources of uncertainties called the data uncertainties. If they are too large, relatively to the model error, they could bias the estimation of the loss function given in Eq. (3.3) causing the hybrid model to overfit on these data and to lack generalisation to other initial conditions. To mitigate this problem, the time series $\mathbf{x}_{0:K}^{\mathrm{a}}$ estimated by DA is filtered using a simple low-pass filter (a rolling mean) producing a smoothed time series $\mathbf{x}_{0:K}^{\mathrm{s}}$. This filter is expected to correct for data uncertainty provided that the observation error is uncorrelated in time, and thus contains high-temporal frequencies. On the other hand, this filtering could remove the fastest scales of the unresolved part of the model. Although it has been assumed in the linear superposition assumption that these high-frequencies can be neglected, we do not know a priori to which extent, which can lead to possibly hamper the forecast skill. The scale separation between the model error acting on long time scales and the initial errors acting on faster scales has been used in previous studies either to estimate the model error in DA [18] or to improve the forecast of a NN model [11]. Finally, note that the filtering can be adapted separately for the fast and the slow variables contained in $\mathbf{x}_k$. This is, for instance, the case in coupled atmosphere-ocean models, and is addressed in section 5.

Algorithm 1 Summary of the algorithm used to determine the hybrid model

```
Input: Observations $\mathbf{y}_{0:K}$,
    truncated model $\mathcal{M}^{\mathrm{r}}$,
    NN architecture $g(\mathbf{x}, \boldsymbol{\theta})$.
Output: state vector estimation $\mathbf{x}_{0:K}^{\mathrm{s}}$,
    optimal value of $\boldsymbol{\theta}$.
if $\mathbf{y}_{0:K}$ is perfect then
    $\mathbf{x}_{0:K}^{\mathrm{s}} = \mathbf{y}_{0:K}$
else
    Use a DA algorithm (e.g. EnKF-N) to estimate the state vector series $\mathbf{x}_{0:K}^{\mathrm{a}}$
    Filter the components (or a subset of components) of $\mathbf{x}_{0:K}^{\mathrm{a}}$ using a low-pass filter to produced the smoothed field $\mathbf{x}_{0:K}^{\mathrm{s}}$
end if
compute the target for the NN: $\boldsymbol{\epsilon}_k^{\mathrm{m}} = (1/N_c)(\mathbf{x}_{k+1}^{\mathrm{s}} - \mathcal{M}^{\mathrm{r}}(\mathbf{x}_k^{\mathrm{s}}, N_c\delta t))$
Determine $\boldsymbol{\theta}$ (training of the NN) using the dataset $(\mathbf{x}_{0:K-1}^{s}; \boldsymbol{\epsilon}_{0:K-1}^{\mathrm{m}})$
return $(\mathbf{x}_{0:K}^{\mathrm{s}}, \boldsymbol{\theta})$
```

In both perfect and imperfect observations cases, the loss function $L^{\mathrm{a}}(\boldsymbol{\theta})$ is minimised using a standard NN training procedure: if $g(\mathbf{x}_k, \boldsymbol{\theta})$ is represented by a NN, $\mathbf{x}_k$ as the inputs and $\boldsymbol{\theta}$ as weight, the problem is equivalent to a supervised regression problem in which the targeted output of the neural network is $\boldsymbol{\epsilon}_k^{\mathrm{m}} = (1/N_c)(\mathbf{x}_{k+1} - \mathbf{x}_k^{\mathrm{r}})$. The algorithm is summarised in Algorithm 1.

(c) Numerical experiment protocol

The performance of the algorithm is evaluated on twin experiments: a full model $\mathcal{M}$ is used to produce true states, synthetic observations, and to assess the forecast skill of the hybrid model. The truncated model $\mathcal{M}^{\mathrm{r}}$ is obtained by neglecting some components of the true model.

A series of true states $\mathbf{x}_{0:K}$ is produced using the true model after projection in the reduced space. This series is used as perfect observations to obtain the so-called "perfect observation-derived hybrid model". Then, synthetic observations are generated using the definition in Eq. (2.8). Here, it is assumed that the observation operator and the observation error statistics are perfectly known. This assumption is not needed for the functioning of the proposed algorithm and could be relaxed at a future stage. Nonetheless, the assumption is done here to simplify the interpretation of the results so the focus is on the truncated model error. Furthermore, the observation error statistics can also be estimated within the DA process itself (see [40] for a review of such methods), and thus be integrated into our combined ML-DA method.

The observations generated using Eq. (2.8) are used in Algorithm 1 to produce the so-called "DA-derived hybrid model". Note that the perfect observation-derived hybrid model benefits from the optimal information at a given time step (the state is completely observed without noise). Given the conditions stated in section 2(b), it is expected to be the best possible model and represents the benchmark against which we will test the DA-derived hybrid model.

(d) Evaluation metrics

The skill of each model is evaluated by running an ensemble of $N^{\mathrm{f}} = 20$ forecasts for N^{f} different initial conditions and for a time period τ: $\mathbf{x}^{\mathrm{f}(l)}(\tau)$ $(l = 1, \cdots, N^{\mathrm{f}})$. The N^{f} different initial conditions are chosen on the attractor of the true model (by running the model long enough before setting the initial conditions), independent of the values of the time series $\mathbf{x}_{0:K}$ used to perform the DA and the NN training. This ensemble of forecast constitutes the so-called test dataset, as it is not used to optimise nor to tune the algorithm. If tuning the algorithm is needed, the original time series $\mathbf{x}_{0:K}$ can be split in a training part (used to optimise the NN) and a validation part (used to tune the NN algorithm).

As evaluation metric we will use the relative root mean square error (R-RMSE)

$$\text{R-RMSE}(\tau, n) = \sqrt{\frac{1}{N^{\mathrm{f}}} \sum_{l=1}^{N^{\mathrm{f}}} \frac{1}{2V_n(\mathbf{x}^{\mathrm{t}})} \left(x_n^{\mathrm{f}(l)}(\tau) - x_n^{\mathrm{t}(l)}(\tau) \right)^2}, \tag{3.4}$$

where $x_n^{\mathrm{f}(l)}(\tau)$ (resp. $x_n^{\mathrm{t}(l)}(\tau)$) is the forecast from the hybrid or the truncated model (resp. the true model) at time τ for the n-th component of $\mathbf{x}$ and for the sample l corresponding to a simulation for one particular initial condition. $V_n(\mathbf{x}^{\mathrm{t}})$ is the n-th component of the variance over the time dimension of the true model forecast time series $\mathbf{x}^{\mathrm{t}}$. Note that if the truncated and/or the hybrid models have the same variability as the true one (i.e. the same variance in time), the R-RMSE converges to 1.

4. Application to the two-scale Lorenz model

(a) Description of the model

The two-scale Lorenz model [27], hereafter L2S, is given by the following set of ODEs:

$$\begin{aligned}
\frac{\mathrm{d}x_n}{\mathrm{d}t} &= \psi_n^{+}(\mathbf{x}) + F - \frac{c}{b} \sum_{m=0}^{9} u_{m+10n} \\
\frac{\mathrm{d}u_m}{\mathrm{d}t} &= \frac{c}{b} \psi_m^{-}(b\mathbf{u}) + h \frac{c}{b} x_{m/10}, \\
\psi_n^{\pm}(\mathbf{x}) &= x_{n\mp1}(x_{n\pm1} - x_{n\mp2}) - u_n,
\end{aligned} \tag{4.1}$$

where $n = 0, \cdots, N_x - 1$ $(N_x = 36)$ and $m = 0, \cdots, N_u - 1$ $(N_u = 360)$. The indices n are periodic, e.g., $x_0 = x_{N_x}$. The values chosen for the parameters are the same as in [28]: the time scale ratio c is set to 10, the space-scale ratio $b = 10$, the coupling $h = 1$ and the forcing $F = 10$. Time t is expressed in model time unit, denoted MTU hereafter.

This set of two-scale ODEs, considered as the true model, is integrated using a fourth-order Runge-Kutta scheme with a time step of 0.005 MTU. The ODEs describing the evolution of $\mathbf{x}$ only represent the truncated model and are obtained by setting the coupling c to 0. It is integrated using a fourth-order Runge-Kutta scheme with a time step of 0.01 MTU.

(b) Setup of the reference experiment

A so-called "reference experiment" is defined in this section. The true model is integrated over approximately 1500 MTU after a spinup of 3 MTU to produce the true state, on which observational noise is added. The EnKF-N is used to assimilate these observations using a large number $N = 50 > N_x$ of ensemble members to reduce sampling errors. A noise is added to the state vector after each forecast to approximately account for the model error due to the model being truncated. It helps to avoid filter divergence and can be seen as additive inflation. This step is necessary given that, due to model error, the forecast ensemble would be otherwise under-dispersive. The noise is assumed Gaussian with zero mean and standard deviation $\boldsymbol{\sigma}^{\mathrm{m}} = \mathbf{0.06}$ optimised after tuning experiments (not shown here). In this reference experiment, the analysis obtained from the DA is not filtered, yet the sensitivity to the filtering of the DA analysis is studied in section 4(d).

The last step of the algorithm is to train a neural network to emulate the unresolved part of the model on the 1500 MTU time series produced by DA. The NN architecture is composed of convolutional layers (denoted “conv.” in Table 1) where the non-linear activation function is a hyperbolic tangent (denoted "tanh"). Some additional parameters have been added, mainly to regularise the training: a batchnorm layer at the input layer, which normalises the training batch, and a L2-regularisation term on the parameters of the last layer. The parameters of the NN are optimised using the "RMSprop" [29] optimiser over 100 epochs. For each epoch, batches of 33 training examples are used to optimise the weights, until all the examples are consumed: this is a standard stochastic minimisation procedure [30]. Full details on the reference experiment are given in Table 1, in the column labelled L2S.

(c) Results

The terminology of the experiments described in the following is recalled in table 2. In Figure 1, both the true model and the DA-derived hybrid model (based on the reference experiment described in section 4(b)) are initialised from an initial condition on the attractor, chosen to be independent of the training set $\mathbf{x}_{0:K}$. The true and the hybrid model are run over 5 MTU, and their difference is displayed. It can be noticed that both runs are very close until 2 MTU and that the hybrid model has predictive skill until 3-4 MTU for this particular set of initial conditions. Note that the Lyapunov time of the truncated model is 0.72 [33], meaning that the hybrid model provides accurate forecasts until 3 Lyapunov times.

In Figure 2, the R-RMSE is averaged over 20 members corresponding to 20 initial conditions and also across all the 36 components of $\mathbf{x}$. R-RMSE is displayed as a function of time. Several densities of observations have been considered: if $N_y = 36$ the full state is observed in the reduced space at each observation time. If $N_y < 36$, $\mathbf{H}_k$ is a sub-sampling operator that draws randomly N_y values from the state following a uniform distribution changing the observation locations at each time step.

Results shown in Figure 2 (left panel) confirms that the DA-derived hybrid model has a predictive skill, significantly better than the truncated model until 4 MTU. The effect of reduced observation density is minor: the skill of the various hybrid models’ forecasts is very similar. This shows the algorithm efficiency in handling sparse data to accurately train a NN model. This is a key strength of our method; most of the other approaches that parametrise a part of the model using ML assume dense observations, e.g. [3,4,7] (similarly to our perfect observation case).

Table 1. Settings of the numerical experiments with the L2S “reference experiment” and with MAOOAM.

		L2S	MAOOAM	
Parameter	Symbol	Value	Value	Note
climatological std	$\boldsymbol{\sigma}^{\mathrm{hf}}$	NA		calculation in 5(b).
Model parameter				
Size of the state	N_x	36	36	
Integration time step	δt	0.01 MTU	1.6 min	
Integration time	T	1500 MTU	62 years.	
Imperfect observation setting				
Standard deviation	$\boldsymbol{\sigma}^{\mathrm{o}}$	0.1	$0.1\boldsymbol{\sigma}^{\mathrm{hf}}$	Ocean filtered.
Observation operator	H	$\mathcal{I}_{N_x}$	$\mathcal{I}_{N_x}$	Identity matrix
Time sampling	Δt	0.05 MTU	27 hours	
Data assimilation				
DA algorithm	EnKF-N			
Ensemble size	N	50	50	
Model additive noise	$\boldsymbol{\sigma}^{\mathrm{m}}$	0.06	$10^{-3}\boldsymbol{\sigma}^{\mathrm{hf}}_{1:20}$	Atmosphere only
Low-pass filtering size		No	55 days	Ocean only.
Neural Network				
		L2S	MAOOAM	
Type of Layer 1		Batchnorm	Batchnorm	[31]
Type of Layer 2		conv.	dense	
Size of Layer 2		43	100	
Activation of Layer 2		tanh	ReLU	[32]
Filter size of Layer 2		5	-	
Type of Layer 3		conv.	dense	
Size of Layer 3		28	50	
Filter size of Layer 2		1	-	
Activation of Layer 3		tanh	ReLU	
Type of Layer 4		conv.	dense	
Size of Layer 4		36	36	
Activation of Layer 4		Linear	Linear	
L2 regularisation		0.07	10^{-4}	
Optimiser		RMSprop	RMSprop	[29]
Number of epochs		100	100	
batch size		33	128	

Table 2. Numerical experiments terminology

Reference experiment	Setup defined in table 1 (column L2S).
DA-derived hybrid model	NN trained with data assimilation reanalysis obtained from noisy and sparse observations.
Perfect-observation-derived hybrid model	NN trained with perfect observations.

(d) Sensitivity studies

In Figure 2 (middle and right panels), the forecast sensitivity to different parameters is studied using the R-RMSE at a lead time $\tau = 2$MTU, averaged over 20 simulations, and over all components of $\mathbf{x}$. The middle panel shows the sensitivity to the observation sampling frequency. For the perfect observation-derived and DA-derived hybrid models, the forecast skill is sensitive to the value of Δt. The forecast skill is significantly degraded for higher values of Δt. This is

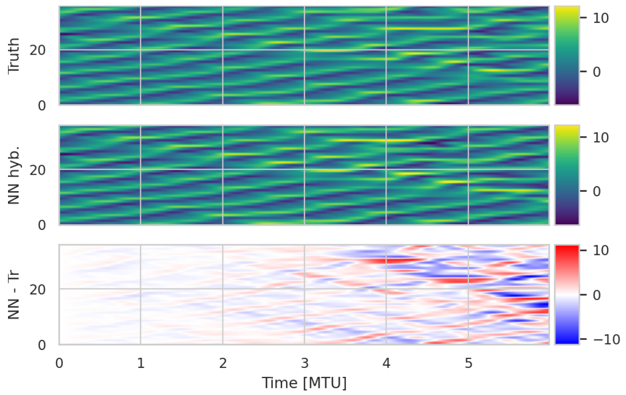

Figure 1. Hovmøller plot of the true model (upper panel) and of the DA-derived hybrid model (middle panel) for the same particular initial conditions over 5 MTU. The bottom panel shows the difference between the two simulations. The setup of the experiment is detailed in Table 1.

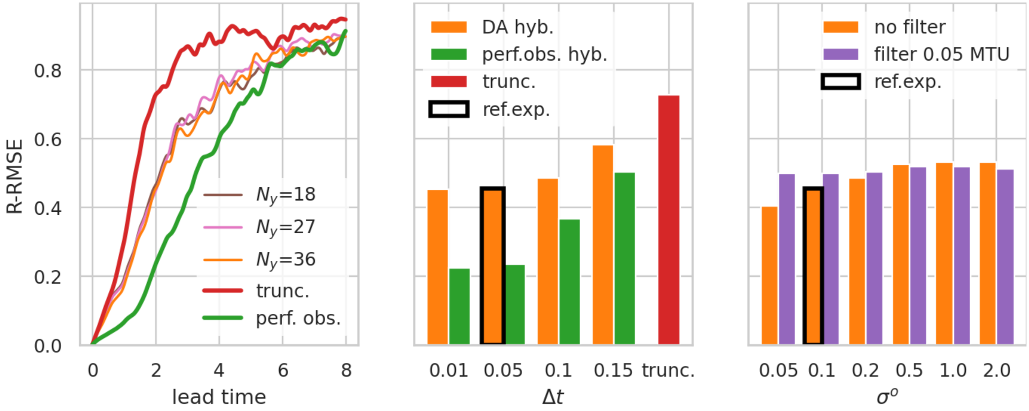

Figure 2. Left: R-RMSE versus time for the perfect observations-derived hybrid model (green), the truncated model (red) and the DA-derived hybrid model (other colours) for 3 different densities of observation. Observation standard deviation, $\sigma^o = 0.1$, the time interval, $\Delta t = 0.05\ MTUs$, are as in the reference experiment; thus for $N_y = 36$ we retrieve the reference experiment. **Middle**: R-RMSE at a lead time $\tau = 2$ MTUs for the DA-derived hybrid models (orange) and the perfect observation-derived hybrid models (green) for different observation sampling time Δt as well as the truncated model (red). **Right**: R-RMSE for the hybrid models trained with no filtering of DA analysis (orange) and with a 0.05 MTU window filter (purple) for different observation error standard deviation σ^o. Black contour indicates the reference experiment conditions described in Table 1.

probably due to the violation of the linear superposition assumption for high values of Δt so that the coupling between the resolved part and the unresolved part of the model, as well as the effect of non-linearity of the unresolved part, are no longer negligible.

The right panel of Figure 2 examines the impact of the observational noise. The result of the reference setting (where the analysis is not filtered before training the NN) is compared with the case of filtering with a rolling mean of 0.05 MTU (*i.e.* 5 time steps). Without filtering, the forecast skill deteriorates as the observation noise increases. Filtering the signal for small noise deteriorates the forecast skill too. This means that some source of predictability lies in the fast scales of this model (which confirms results from the middle panel, when it appears that shorter time sampling for observation improves the forecast skill). This small temporal scale variability is damped by the filter, but also by the increase of the observational noise that tends to add randomness on all scales, including the small ones. In this case, except for very strong noise, filtering does not seem to improve drastically the forecast skill. Notably, even a strong noise in the data has only a very small influence on the forecast skill of the hybrid model: when the noise on the observation is multiplied by a factor 20, the error in the forecast at $t_0 + 2$MTU is only multiplied by a factor 1.3.

From the results on the 2-scales Lorenz model, we conclude that the algorithm is robust against varying data spatial density, but is sensitive to their temporal distribution. Also, filtering the analyses obtained from DA may appear useful for slow processes but can deteriorate the results by filtering significant fast processes.

5. Application to a low-order coupled ocean-atmosphere model

(a) Description of the model

We consider here the Modular Arbitrary-Order-Ocean-Atmosphere Model (MAOOAM) introduced by [34]. MAOOAM has 3 layers (2 for the atmosphere and 1 for the ocean) and is a reduced-order quasi-geostrophic model resolved in the spectral space. Its state is composed of n_a modes of the atmospheric barotropic streamfunction $\psi_{a,i}$ and the atmospheric temperature anomaly $\theta_{a,i}$ respectively, plus n_o modes of the oceanic streamfunction $\psi_{o,j}$ and the oceanic temperature anomaly $\theta_{o,j}$ respectively. The total number of variables is $N_x = 2n_a + 2n_o$. We consider two versions of MAOOAM: the true model with dimension $N_z = 56$ ($n_a = 20$,$n_o = 8$) corresponding to the configuration "atm. $2x$-$2y$ oc. $2x$-$4y$" in [34] and the truncated model with $N_x = 36$ ($n_a = 10$,$n_o = 8$) corresponding to the configuration atm. "$2x$-$4y$ oc. $2x$-$4y$" in [34]. The truncated model is missing 20 high-order atmospheric variables (10 for the streamfunction and 10 for the temperature anomaly). Thus the truncated model does not resolve the atmosphere-ocean coupling related to these high order atmospheric modes.

The true model is used to generate synthetic observations. The forecast skill and the long-term properties of the truncated and the hybrid models will be evaluated by inspecting 3 crucial model variables, called *key variables*, that are $\psi_{o,2}$, $\theta_{o,2}$ and $\psi_{a,1}$ (*i.e.* the second components of ocean streamfunction and temperature and the first component of the atmospheric streamfunction). They account for 42%, 51%, and 18% respectively of the variability of a reanalysis of 2-dimensional fields [35], and have been already used in previous studies (*e.g.* [36]). MAOOAM has also been recently used to study coupled data assimilation methods [37,38]. Unsurprisingly, in MAOOAM the ocean variables are considered the slow ones while the atmospheric variables are the fast ones.

(b) Experimental setup

We will express time in real time units (minutes, hours, days, ...) but, in practice, the model time is non-dimensional. Consequently, the dimensioned time values presented hereafter are not round numbers.

Given the diverse time scales and amplitudes of the MAOOAM variables, the noise parameters are all scaled on a climatological standard deviation of high frequencies $\boldsymbol{\sigma}^{\mathrm{hf}} \in \mathbb{R}^{N_x}$, which is defined as the temporal standard deviation of the state vector after filtering out slow variations of a period longer than 1 month. This high-pass filter is carried out by subtracting the 1-month running average.

The parameters of the experiments are presented in Table 1, in the column labelled MAOOAM. The true model is integrated over approximately 62 years after a spinup of $30,000$ years, in the same configuration as in [34]. In all experiments with MAOOAM, the state is fully observed every 27 hours ($\Delta t = 27$ hours) (corresponding to $N_c = 1,000$). A small modification was made to the observations from Eq. (2.8) to account for the fact that observations of the ocean are not at the same scale as those of the atmosphere: before being assimilated, instantaneous ocean observations are averaged over a 55 days rolling period centred at the analysis times. The EnKF-N is used as DA algorithm. The noise on the model forecast is added only to the atmospheric variables with a standard deviation of $\boldsymbol{\sigma}^{\mathrm{m}} = 10^{-3}\boldsymbol{\sigma}^{\mathrm{hf}}_{1:20}$. As mentioned in section 3(b), the analysis obtained from the DA is filtered. The slow processes are expected to occur mainly in the ocean, so only the ocean components of the state vector $\mathbf{x}^{\mathrm{a}}_{0:K}$ are filtered to produce $\mathbf{x}^{\mathrm{s}}_{0:K}$. Differently from the L2S model experiments, filtering the analysis has proven necessary to train the hybrid model using MAOOAM.

The NN-architecture is a simple 3 layers multi-layer perceptrons; see Table 1 for full details. As opposed to the L2S model, the state vector has no locality properties (because it is defined in the spectral space), so the convolutional layers are not applicable (see the discussion about locality in [20]). The training of the NN is performed in the same way as for the L2S experiments.

(c) Results

The forecast skill metrics are presented in Table 3 for the truncated model as well as for the perfect observation-derived and the DA-derived hybrid models. Given the different time scales involved, the forecast lead time of the key atmospheric variable $\psi_{a,1}$ is 1 day whereas the forecast lead time of the two key oceanic variables $\psi_{o,2}$ and $\theta_{o,2}$ is 2 years. It can be seen that both perfect observation-derived and DA-derived hybrid models have superior skill to the truncated model. The improvement is larger for the ocean, with a factor of 2 to 3, and is similar for both hybrid models. Recall that the true model has here the same oceanic variables as the truncated model, so there is no difference in the representation of the pure oceanic processes. The improvement is thus fully due to an enhanced representation of the atmosphere-ocean coupling processes, the hybrid model better representing the interplay between the unresolved fast atmospheric variables and the slow oceanic variables.

The atmospheric key variable is improved to a lesser extent by the two hybrid models, and the perfect observations-derived model is significantly better than the DA derived model. This proves the limited capability of the hybrid model to represent a fast process, a situation further exacerbated in the case of the DA-derived hybrid model, when only noisy and partial data are at disposal. This result was indeed expected given the assumptions made on the unresolved term of the model in section 3(a) when a slow variation of the unresolved term was assumed. The fast processes are also less accurately represented because the sampling rate of the observations (27 hours) is well beyond the atmospheric time scale, and because of the presence of the observation errors (when applied).

In Figure 3, the attractors of the different models are displayed in the phase space defined by two key variable: $\psi_{o,2}$ and $\psi_{a,1}$. A significant difference can immediately be seen between the attractors of the truncated and the true models: the truncated model visits areas of the phase space that are not admitted in the real dynamics. Remarkably, these discrepancies are reduced by the hybrid models both derived from perfect observation and from DA. Some states seem to remain out of the true model attractor, however, but much fewer.

Quantitative characterisation of the attractors is presented in Table 4, which provides the 3 quartiles (including the median) for each key variable. For the truncated model and the hybrid models, the difference of the quartiles is given relatively to the true model. For all the oceanic variables the distribution of the values of both hybrid models is significantly closer to the true distribution than for the truncated model. For the key atmospheric variable $\psi_{a,1}$, only the hybrid model derived from perfect observations shows an improvement. It confirms the conclusion made on the forecast skill that the hybrid model represents well the slow process, in particular oceanic variables in this case, and that the fast processes are not fully retrieved, in particular in case of the DA-derived hybrid model.

6. Conclusion

We have developed a novel method to build a hybrid model consisting of a physics-based truncated model and a data-driven model of the unresolved processes. The approach is based

Table 3. Forecast R-RMSE of hybrid and truncated MAOOAM models

R-RMSE(lead time τ)	$\psi_{o,2}$(2 years)	$\theta_{o,2}$(2 years)	$\psi_{a,1}$(1 day)
Truncated	0.23	0.21	0.36
Perfect obs. hybrid	0.07	0.07	0.23
DA hybrid	0.10	0.06	0.28

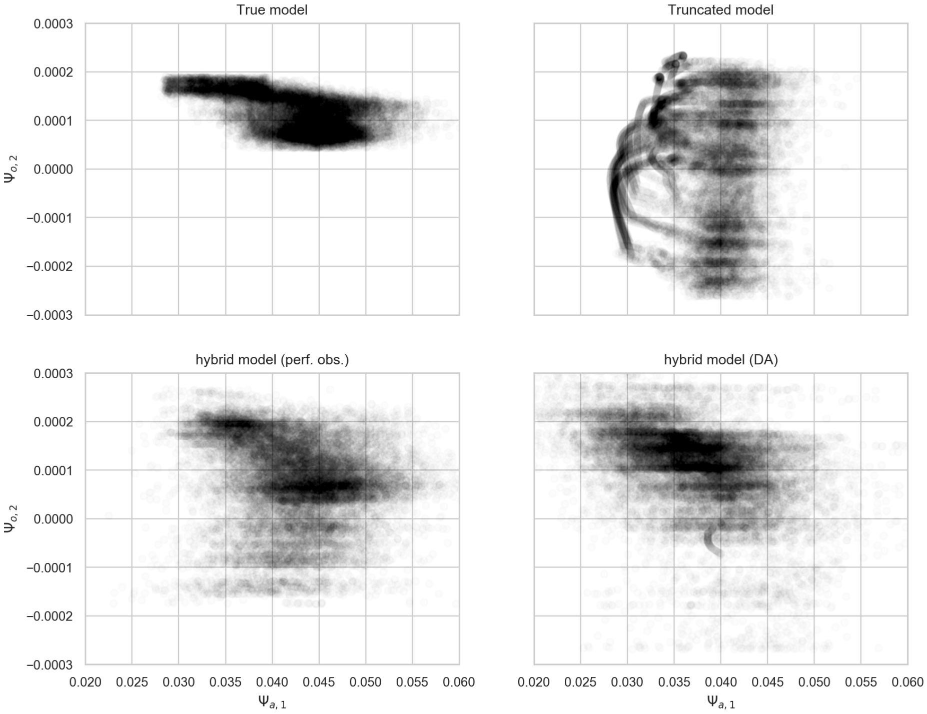

Figure 3. Cross-section of the attractor for two key variables $\psi_{a,1}$ and $\psi_{o,2}$ in the true model (upper left), the truncated model (upper right), the perfect-observation-derived hybrid model (lower left) and the DA-derived hybrid model (lower right).

Table 4. Quartiles of the key variables for the MAOOAM model relative to the true model.

	$\psi_{o,2}$			$\theta_{o,2}$		
	Q1	M	Q3	Q1	M	Q3
True model	$7.8 \cdot 10^{-5}$	$1.1 \cdot 10^{-4}$	$1.5 \cdot 10^{-4}$	$8.2 \cdot 10^{-2}$	$1.2 \cdot 10^{-1}$	$1.4 \cdot 10^{-1}$
Truncated	-229%	-80%	-26%	-22%	-10%	-6%
Perfect obs. hybrid	-55%	-26%	0.6%	7%	-2%	-4%
DA hybrid	-14%	9%	8%	8%	-5%	-0.2%

	$\psi_{a,1}$		
	Q1	M	Q3
True model	$3.9 \cdot 10^{-2}$	$4.3 \cdot 10^{-2}$	$4.6 \cdot 10^{-2}$
Truncated	-12%	-11%	-11%
Perfect obs. hybrid	-0.6%	-2%	0.02%
DA hybrid	-15%	-14%	-11%

on realistic assumptions that only noisy and incomplete observations are available at a lower frequency than the model integration time step.

With a two-scale low-order chaotic system [27], we showed that the hybrid model forecast skill is sensitive to the observation frequency but very robust against high observational noise and sparse spatial distribution. This is probably due to the fact that reduced observation frequencies challenge the validity of the linear superposition assumption more than large observational noise (see the discussion in Appendix of [39]). We then applied the method to the low-order coupled ocean-atmosphere model MAOOAM [34] which contains multiple temporal scales. Forecast skill and global statistics were significantly improved by the hybrid model compared with the truncated model encouraging further studies to high-dimensional and more realistic scenarios. Notably, the hybrid model derived from noisy observations has comparable forecast skill on the oceanic variables to that of the hybrid model derived from perfect observations.

In view of operational systems, it should be noted that the proposed algorithm relies on two existing data assimilation and neural networks training techniques that both scale well in high-dimension (see, *e.g.*, [41] and [42]). In principle, the present algorithm can be applied to larger and more realistic problems. In particular, the fact that the method does not rely on the adjoint of the truncated model is an advantage in terms of code maintenance. However, we foresee some practical challenges: for instance, the computational architecture and the data types used for physics-based numerical models and for machine learning algorithms can be very different (*e.g.* multi-core supercomputers and graphics processing units). Training and running hybrid models efficiently imposes heavy requirements on both the hardware and software and may come with an overhead even if some tools are very promising [43].

The approach presented here can also accommodate the additional representation of the remaining model error (*i.e.* the model error of the hybrid model): it could either be done within the numerical model by parameterising the model error [21], or by training stochastic neural networks [44].

Data Accessibility. Data used in all the experiments are available at `ftp://ftp.nersc.no/reddaml/`. The instructions to download the data, to run all the Lorenz 2 scale application and to reproduce figures and tables of the article can be found at `https://github.com/brajard/reddaml/releases/tag/v1.0`.

Authors' Contributions. JB first proposed the theory, implemented and conducted the numerical experiments. All authors have contributed to the interpretation of the theory and the results as well as the edition of the manuscript. All authors approved the manuscript.

Competing Interests. The authors declare that they have no competing interests.

Funding. JB and LB have been funded by the project REDDA (#250711) of the Norwegian Research Council. JB has also been funded by the project SFE (#2700733) of the Norwegian Research Council. AC has been funded by the UK Natural Environment Research Council award NCEO02004. CEREA is member of Institut Pierre–Simon Laplace (IPSL).

Acknowledgements. The authors are thankful to Patrick Raanes (NORCE, NO) for his support on the use of the data assimilation python platform DAPPER, and to Jonathan Demaeyer (RMI, BE) for the insightful discussions about MAOOAM. Thanks to Joséphine Schmutz (IMT) for his review of the GitHub repository. Many thanks also to the two anonymous reviewers for their valuable comments.

References

1. Merchant CJ, Embury O, Bulgin CE, Block T, Corlett GK, Fiedler E, Good SA, Mittaz J, Rayner NA, Berry D, *et al.* 2019 Satellite-based time-series of sea-surface temperature since 1981 for climate applications.
Scientific data **6**, 1–18.
2. Carrassi A, Bocquet M, Bertino L, Evensen G. 2018 Data assimilation in the geosciences: An overview of methods, issues, and perspectives.
Wiley Interdisciplinary Reviews: Climate Change **9**, e535.
3. O'Gorman PA, Dwyer JG. 2018 Using machine learning to parameterize moist convection: Potential for modeling of climate, climate change, and extreme events.
Journal of Advances in Modeling Earth Systems **10**, 2548–2563.

4. Rasp S, Pritchard MS, Gentine P. 2018 Deep learning to represent subgrid processes in climate models.
Proceedings of the National Academy of Sciences **115**, 9684–9689.
5. Krasnopolsky VM, Fox-Rabinovitz MS, Belochitski AA. 2013 Using ensemble of neural networks to learn stochastic convection parameterizations for climate and numerical weather prediction models from data simulated by a cloud resolving model.
Advances in Artificial Neural Systems **2013**.
6. Bolton T, Zanna L. 2019 Applications of deep learning to ocean data inference and subgrid parameterization.
Journal of Advances in Modeling Earth Systems **11**, 376–399.
7. Brenowitz ND, Bretherton CS. 2018 Prognostic validation of a neural network unified physics parameterization.
Geophysical Research Letters **45**, 6289–6298.
8. Rasp S. 2020 Coupled online learning as a way to tackle instabilities and biases in neural network parameterizations: general algorithms and Lorenz 96 case study (v1. 0).
Geoscientific Model Development **13**, 2185–2185.
9. Brunton SL, Proctor JL, Kutz JN. 2016 Discovering governing equations from data by sparse identification of nonlinear dynamical systems.
Proceedings of the national academy of sciences **113**, 3932–3937.
10. Pathak J, Hunt B, Girvan M, Lu Z, Ott E. 2018 Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach.
Physical review letters **120**, 024102.
11. Faranda D, Vrac M, Yiou P, Pons FME, Hamid A, Carella G, Ngoungue Langue CG, Thao S, Gautard V. 2020 Boosting performance in machine learning of geophysical flows via scale separation.
Working paper or preprint
12. Park DC. 2010 A time series data prediction scheme using bilinear recurrent neural network.
In *2010 International Conference on Information Science and Applications*, pp. 1–7. IEEE.
13. Fablet R, Ouala S, Herzet C. 2018 Bilinear residual neural network for the identification and forecasting of geophysical dynamics.
In *2018 26th European Signal Processing Conference (EUSIPCO)*, pp. 1477–1481. IEEE.
14. Scher S. 2018 Toward data-driven weather and climate forecasting: Approximating a simple general circulation model with deep learning.
Geophysical Research Letters **45**, 12–616.
15. Dueben PD, Bauer P. 2018 Challenges and design choices for global weather and climate models based on machine learning.
Geoscientific Model Development **11**, 3999–4009.
16. de Bezenac E, Pajot A, Gallinari P. 2019 Deep learning for physical processes: Incorporating prior scientific knowledge.
Journal of Statistical Mechanics: Theory and Experiment **2019**, 124009.
17. Nguyen D, Ouala S, Drumetz L, Fablet R. 2019 Em-like learning chaotic dynamics from noisy and partial observations.
arXiv preprint arXiv:1903.10335 .
18. Laloyaux P, Bonavita M, Dahoui M, Farnan J, Healy S, Hólm E, Lang S. 2020 Towards an unbiased stratospheric analysis.
Quarterly Journal of the Royal Meteorological Society .
19. Bonavita M, Laloyaux P. 2020 Machine learning for model error inference and correction.
Earth and Space Science Open Archive p. 36.
20. Brajard J, Carassi A, Bocquet M, Bertino L. 2020 Combining data assimilation and machine learning to emulate a dynamical model from sparse and noisy observations: a case study with the Lorenz 96 model.
Journal of Computational Science **44**, 101171.
21. Bocquet M, Brajard J, Carrassi A, Bertino L. 2020 Bayesian inference of chaotic dynamics by merging data assimilation, machine learning and expectation-maximization.
Foundations of Data Science **2**, 55–80.
22. Tian X, Zhang H, Feng X, Xie Y. 2018 Nonlinear least squares en4dvar to 4denvar methods for data assimilation: Formulation, analysis, and preliminary evaluation.
Monthly Weather Review **146**, 77–93.

23. Bocquet M, Raanes PN, Hannart A. 2015 Expanding the validity of the ensemble Kalman filter without the intrinsic need for inflation.
Nonlinear Processes in Geophysics **22**, 645.
24. Raanes PN, *et al.* 2018.
nansencenter/dapper: Version 0.8.
25. Carrassi A, Vannitsem S. 2011 Treatment of the error due to unresolved scales in sequential data assimilation.
International Journal of Bifurcation and Chaos **21**, 3619–3626.
26. Mitchell L, Carrassi A. 2015 Accounting for model error due to unresolved scales within ensemble Kalman filtering.
Quarterly Journal of the Royal Meteorological Society **141**, 1417–1428.
27. Lorenz EN. 2005 Designing chaotic models.
Journal of the atmospheric sciences **62**, 1574–1587.
28. Bocquet M, Brajard J, Carrassi A, Bertino L. 2019 Data assimilation as a learning tool to infer ordinary differential equation representations of dynamical models.
Nonlinear Processes in Geophysics **26**, 143–162.
29. Hinton G, Srivastava N, Swersky K. 2012.
Neural networks for machine learning lecture 6a overview of mini-batch gradient descent.
30. Bottou L. 2010 Large-scale machine learning with stochastic gradient descent.
In *Proceedings of COMPSTAT'2010*, pp. 177–186. Springer.
31. Ioffe S. 2017 Batch renormalization: Towards reducing minibatch dependence in batch-normalized models.
In *Advances in neural information processing systems*, pp. 1945–1953.
32. Glorot X, Bordes A, Bengio Y. 2011 Deep sparse rectifier neural networks.
In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 315–323.
33. Carlu M, Ginelli F, Lucarini V, Politi A. 2018 Lyapunov analysis of multiscale dynamics: the slow bundle of the two-scale Lorenz 96 model.
arXiv preprint arXiv:1809.05065 .
34. De Cruz L, Demaeyer J, Vannitsem S. 2016 The modular arbitrary-order ocean-atmosphere model: MAOOAM v1.0.
Geoscientific Model Development **9**, 2793–2808.
35. Vannitsem S, Ghil M. 2017 Evidence of coupling in ocean-atmosphere dynamics over the north atlantic.
Geophysical Research Letters **44**, 2016–2026.
36. Demaeyer J, Vannitsem S. 2017 Stochastic parametrization of subgrid-scale processes in coupled ocean–atmosphere systems: benefits and limitations of response theory.
Quarterly Journal of the Royal Meteorological Society **143**, 881–896.
37. Penny S, Bach E, Bhargava K, Chang CC, Da C, Sun L, Yoshida T. 2019 Strongly coupled data assimilation in multiscale media: Experiments using a quasi-geostrophic coupled model.
Journal of Advances in Modeling Earth Systems **11**, 1803–1829.
38. Tondeur M, Carrassi A, Vannitsem S, Bocquet M. 2020 On temporal scale separation in coupled data assimilation with the ensemble Kalman filter.
Journal of Statistical Physics **179**, 1161–1185.
39. Bocquet M, Carrassi A. 2017 Four-dimensional ensemble variational data assimilation and the unstable subspace.
Tellus A: Dynamic Meteorology and Oceanography **69**, 1304504.
40. Tandeo P, Ailliot P, Bocquet M, Carrassi A, Miyoshi T, Pulido M, Zhen Y. 2020 A Review of Innovation-Based Methods to Jointly Estimate Model and Observation Error Covariance Matrices in Ensemble Data Assimilation.
Monthly Weather Review , .
41. Sakov P, Counillon F, Bertino L, Lisæter K, Oke P, Korablev A. 2012 Topaz4: an ocean-sea ice data assimilation system for the north atlantic and arctic.
Ocean Science Discussions **9**.
42. LeCun Y, Bengio Y, Hinton G. 2015 Deep learning.
nature **521**, 436–444.
43. Ott J, Pritchard M, Best N, Linstead E, Curcic M, Baldi P. 2020 A fortran-keras deep learning bridge for scientific computing.
arXiv preprint arXiv:2004.10652 .

44. Gal Y, Ghahramani Z. 2016 Dropout as a Bayesian approximation: Representing model uncertainty in deep learning.
In *international conference on machine learning*, pp. 1050–1059.